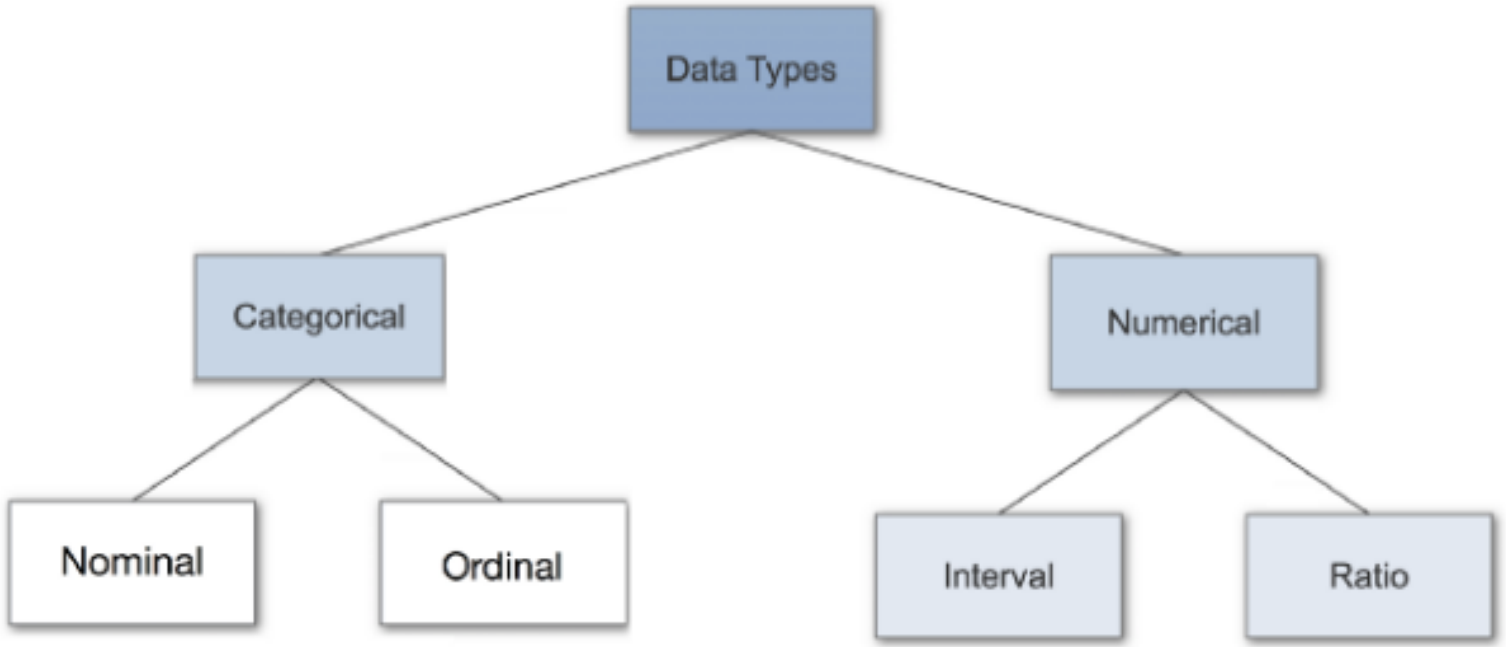
TYPE DATA, MISSING VALUE, OUTLIER


Data Types
Categorical
Numerical
Nominal
Ordinal
Interval
Ratio

TYPE DATA

DATA KATEGORIKAL (QUALITATIVE)

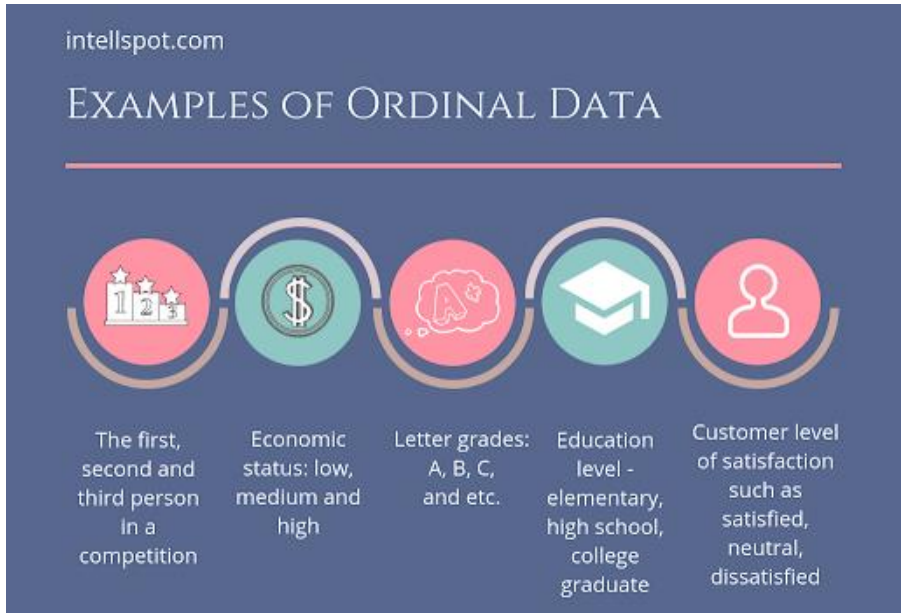
- **Data Nominal**

What is your gender?
- M – Male
- F – Female

What is your hair color?
- 1 – Brown
- 2 – Black
- 3 – Blonde
- 4 – Gray
- 5 – Other

Where do you live?
- A – North of the equator
- B – South of the equator
- C – Neither: In the international space station

- **Data Ordinal**



DATA NUMERIKAL (QUANTITATIVE)

Data Numerik terbagi dua :

- Data Diskrit

  Variabel hasil dari penghitungan, misal : jumlah anak, jumlah pasien dll

- Data kontinyu

  Hasil dari pengukuran, misal : tekanan darah, Hb dll

https://www.slideshare.net/GraciaConsuellaConsuella/analisa-data-amp-uji-statistik

MISSING VALUES

- Missing completely at random (MCAR)
- Missing at random (MAR)
- Missing not at random (MNAR)
- Structurally Missing Data

MISSING COMPLETELY AT RANDOM (MCAR)

- data survei menggunakan subset pertanyaan acak dari daftar yang telah ditentukan sebelumnya.
- data yang diukur dari sensor yang tidak konsisten yang menghasilkan nilai yang hilang

A	B	C	D
	1	0	1
1	0		1
0		1	0
1	0		1
0	0	1	

MISSING AT RANDOM (MAR)

- sensor yang melewatkan pengukuran dalam menit tertentu tetapi merekam data di menit sebelumnya dan menit berikutnya.

A	B	C	D
1	1	0	
1	0	0	
0	1	1	
1	0	0	1
0	0	1	1

MISSING NOT AT RANDOM (MNAR)

- Missing yang diketahui Tetapi tidak dapat diprediksi
- orang dalam kelompok usia/pendapatan tertentu menolak untuk menjawab berapa banyak kendaraan atau rumah yang mereka miliki

A	B	C	D
1	1	0	1
1	0	0	1
0	1		0
1	0	0	1
0	0		1

STRUCTURALLY MISSING DATA

- data yang hilang dengan sengaja. Misalnya, survei yang meminta penghasilan dari pekerjaan akan memiliki nilai missing bagi mereka yang tidak memiliki pekerjaan.

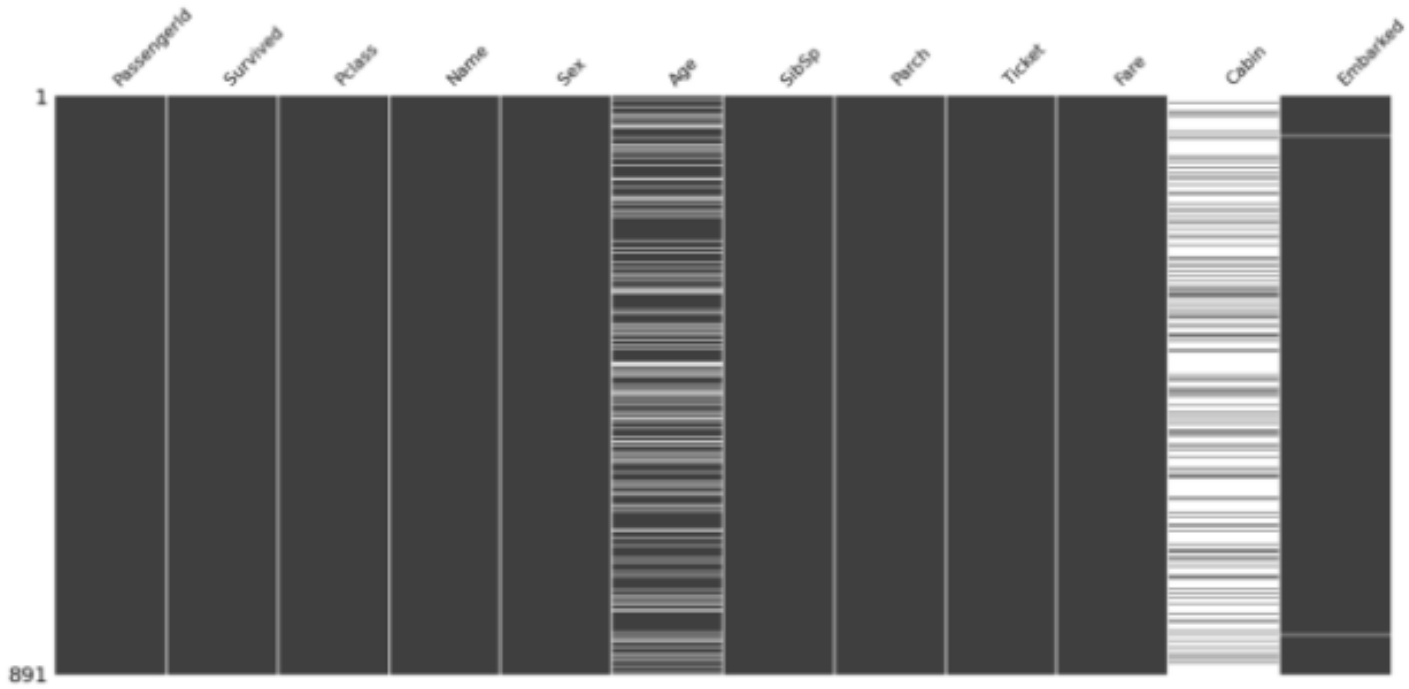
A	B	C	D
1	1	0	1
1	0	0	1
0	1	1	0
1	0	0	1
0			

HANDLING MISSING VALUES

- Deleting Rows with missing values
- Impute missing values for continuous variable
- Impute missing values for categorical variable
- Other Imputation Methods

DELETING ROWS WITH MISSING VALUES

```
msno.matrix(data)
```




```
print(data.isnull().sum())
print(data.shape)
```

```
PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age            177
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
(891, 12)
```

```
data.dropna(inplace=True)
print(data.isnull().sum())
print(data.shape)
```

```
PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Cabin          0
Embarked       0
dtype: int64
(183, 12)
```

Pro & Kontra:
menciptakan model yang bagus
Kehilangan banyak informasi

IMPUTE MISSING VALUES FOR CONTINUOUS VARIABLE

- Kolom yang memiliki data numerik, bisa diganti data yang missing dengan mean, median, modus

```
[10] data["Age"][:20]

0     22.0
1     38.0
2     26.0
3     35.0
4     35.0
5      NaN
6     54.0
7      2.0
8     27.0
9     14.0
10     4.0
11    58.0
12    20.0
13    39.0
14    14.0
15    55.0
16     2.0
17     NaN
18    31.0
19     NaN
Name: Age, dtype: float64
```

```
data["Age"] = data["Age"].replace(np.NaN, data["Age"].mean())
print(data["Age"][:20])

0     22.000000
1     38.000000
2     26.000000
3     35.000000
4     35.000000
5     29.699118
6     54.000000
7      2.000000
8     27.000000
9     14.000000
10     4.000000
11    58.000000
12    20.000000
13    39.000000
14    14.000000
15    55.000000
16     2.000000
17    29.699118
18    31.000000
19    29.699118
Name: Age, dtype: float64
```

Pro:

- Mencegah kehilangan data yang akibat penghapusan baris atau kolom
- Bekerja dengan baik dalam ukuran data yang kecil dan mudah diimplementasikan.

Kontra:

- Bekerja hanya dengan variable kontinyu numerik
- Menyebabkan data leakage

IMPUTE MISSING VALUES FOR CATEGORICAL VARIABLE

- Ketika kolom berisi kategorikal data, missing values bisa direplace dengan data yang paling banyak banyak frekuensinya. Tetapi jika missing valuenya sangat besar, maka bisa diganti dengan kategori baru

```
[12] data.isnull().sum()

PassengerId      0
Survived         0
Pclass           0
Name             0
Sex              0
Age              0
SibSp            0
Parch            0
Ticket           0
Fare             0
Cabin          687
Embarked         2
dtype: int64
```

```
[13] data["Cabin"] = data["Cabin"].fillna('U')

[14] data.isnull().sum()

PassengerId    0
Survived       0
Pclass         0
Name           0
Sex            0
Age            0
SibSp          0
Parch          0
Ticket         0
Fare           0
Cabin          0
Embarked       2
dtype: int64
```

Pro:

- Mencegah kehilangan data yang akibat penghapusan baris atau kolom
- Bekerja dengan baik dalam ukuran data yang kecil dan mudah diimplementasikan.
- Mencegah hilang data dengan menambahkan kategori unik

Kontra:

- Bekerja hanya dengan variable data kategori
- Menambahkan fitur baru mungkin menyebabkan model kurang bagus

OTHER IMPUTATION METHODS

- Last observation carried forward (LOCF) method

```
data["Age"] = data["Age"].fillna(method='ffill')
```

- interpolation of the variable

```
data["Age"] = data["Age"].interpolate(method='linear', limit_direction='forward', axis=0)
```

SINGLE IMPUTAION

Serial	Gender	Income
1	Female	100
2	Female	NA
3	Male	100
4	Female	300
5	Male	NA
6	Male	200
7	Female	200

Serial	Gender	Income
1	Female	100
2	Female	180
3	Male	100
4	Female	300
5	Male	180
6	Male	200
7	Female	200

SINGLE IMPUTATION

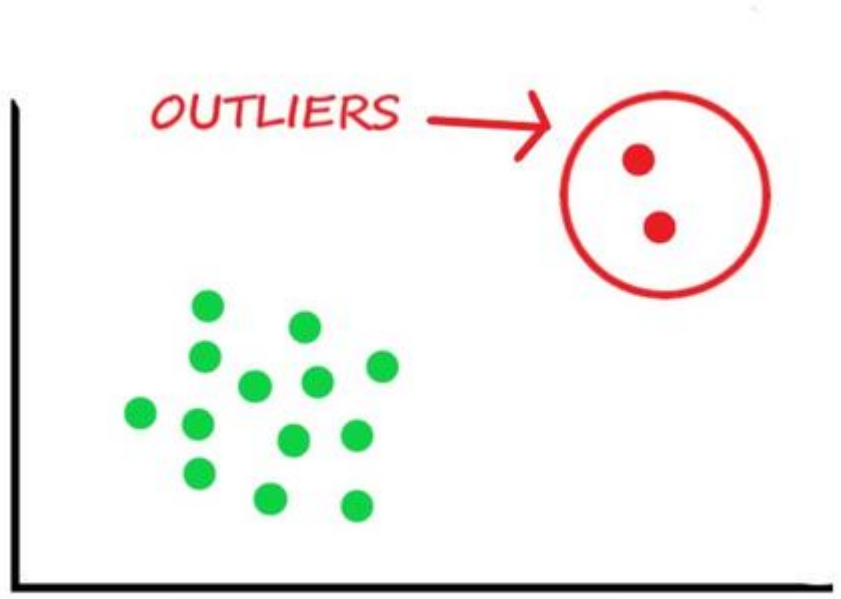
Serial	Age	Death reason
1	60	Covid-19
2	64	NA
3	42	Heart attack
4	67	Covid-19
5	80	NA
6	32	Cancer
7	35	Cancer
8	45	Cancer
9	88	NA
10	33	Heart attack

Shahidul Islam Khan & Abu Sayed Md Latiful Hoque, SICE: an improved missing data imputation technique, Journal of Big Data

OUTLIERS, CHECKING AND HANDLING THEM

OUTLIERS

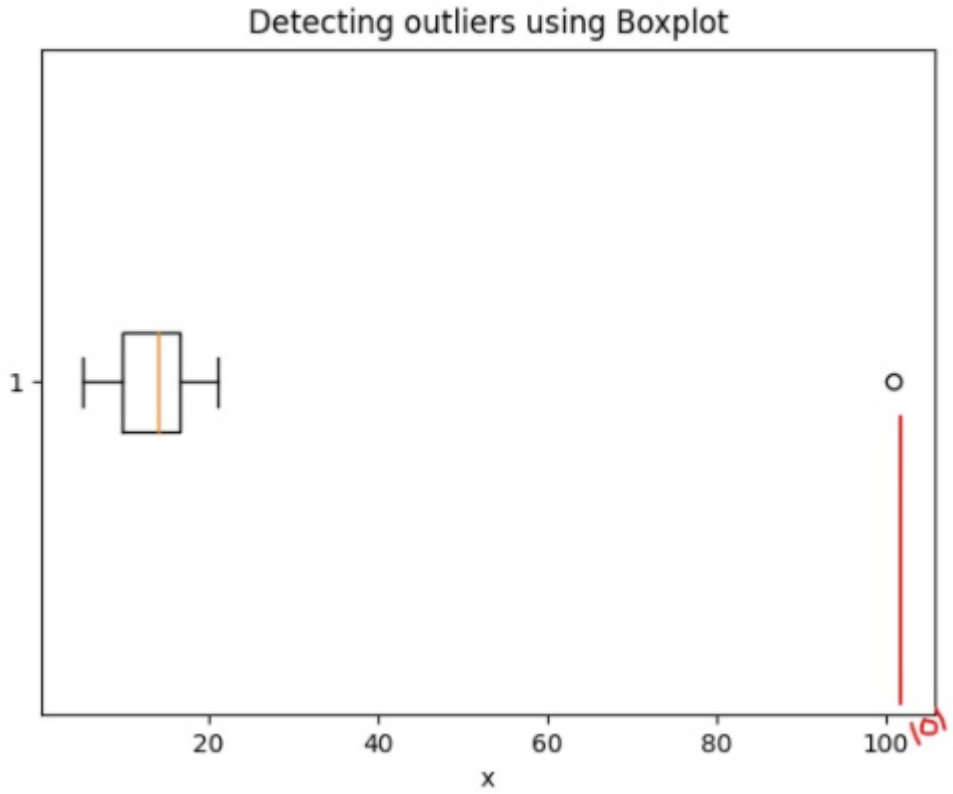
- Sebuah data poin yang terletak jauh dari letak data poin yang lain
- Outlier terjadi karena Variability data atau error data recorded



CHECK OUTLIERS

- Box plot (matplotlib/seaborn)

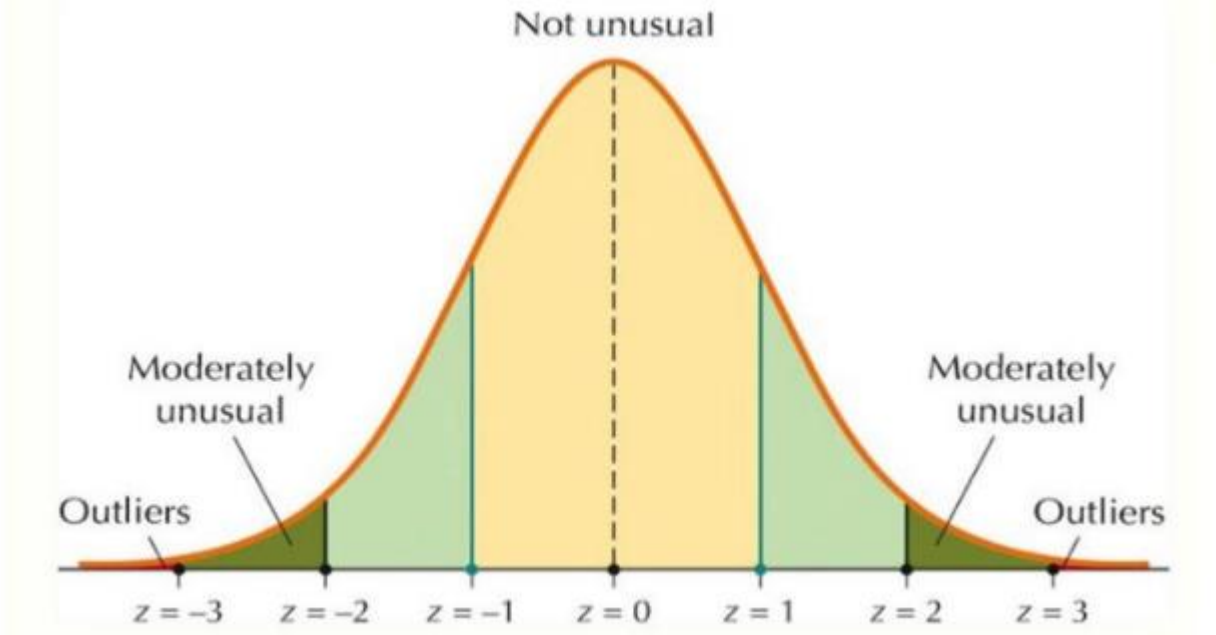
```
import matplotlib.pyplot as plt
plt.boxplot(sample, vert=False)
plt.title("Detecting outliers using Boxplot")
```



CHECK OUTLIERS

- Z-scores

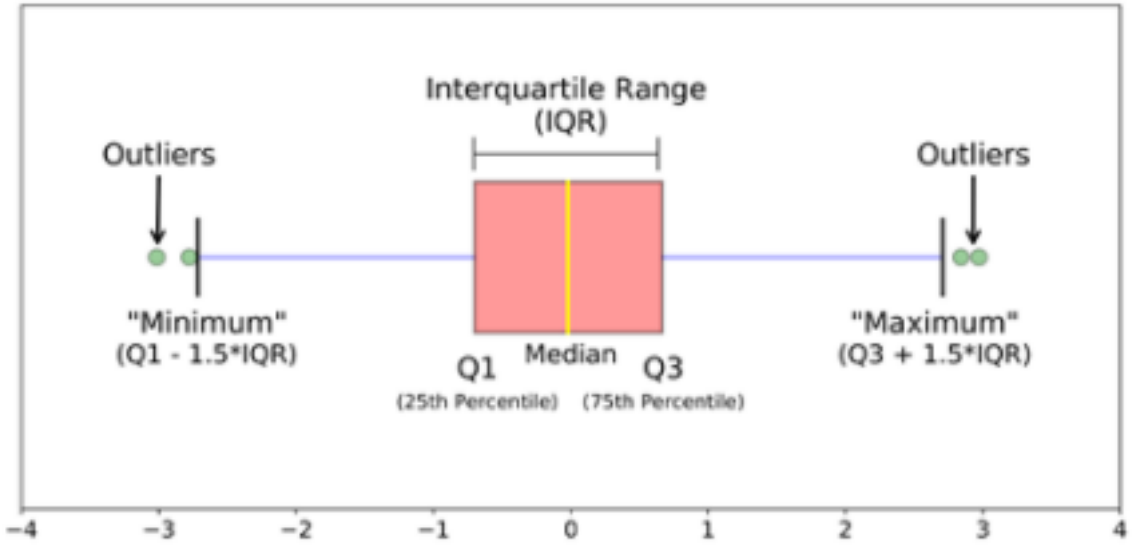
```
import numpy as np
outliers = []
def detect_outliers_zscore(data):
    thres = 3
    mean = np.mean(data)
    std = np.std(data)
    # print(mean, std)
    for i in data:
        z_score = (i-mean)/std
        if (np.abs(z_score) > thres):
            outliers.append(i)
    return outliers# Driver code
sample_outliers = detect_outliers_zscore(sample)
print("Outliers from Z-scores method: ", sample_outliers)
```



CHECK OUTLIERS

- Inter Quartile ranges

```
outliers = []
def detect_outliers_iqr(data):
    data = sorted(data)
    q1 = np.percentile(data, 25)
    q3 = np.percentile(data, 75)
    # print(q1, q3)
    IQR = q3-q1
    lwr_bound = q1-(1.5*IQR)
    upr_bound = q3+(1.5*IQR)
    # print(lwr_bound, upr_bound)
    for i in data:
        if (i<lwr_bound or i>upr_bound):
            outliers.append(i)
    return outliers# Driver code
sample_outliers = detect_outliers_iqr(sample)
print("Outliers from IQR method: ", sample_outliers)
```



HANDLING OUTLIERS

- Trimming/Remove the outliers

```
# Trimming
for i in sample_outliers:
    a = np.delete(sample, np.where(sample==i))
print(a)
# print(len(sample), len(a))
```

- Quantile based flooring and capping

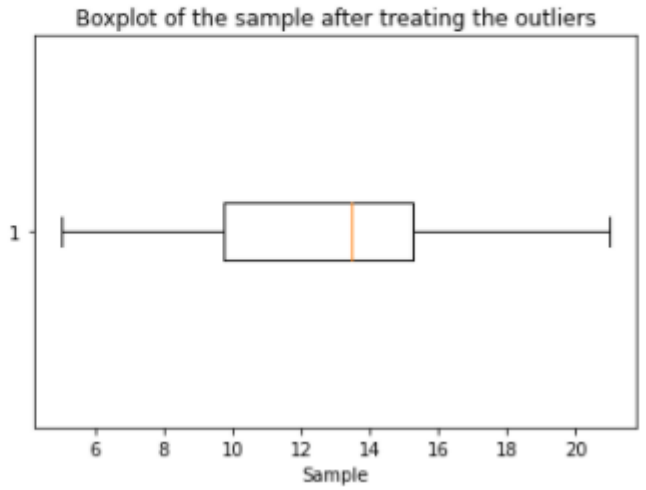
```
# Computing 10th, 90th percentiles and replacing the outliers
tenth_percentile = np.percentile(sample, 10)
ninetieth_percentile = np.percentile(sample, 90)
# print(tenth_percentile, ninetieth_percentile)b = np.where(sample<tenth_percentile, tenth_percentile, sample)
b = np.where(b>ninetieth_percentile, ninetieth_percentile, b)
# print("Sample:", sample)
print("New array:",b)
```

HANDLING OUTLIERS

- Mean/Median imputation

```
median = np.median(sample)# Replace with median
for i in sample_outliers:
    c = np.where(sample==i, 14, sample)
print("Sample: ", sample)
print("New array: ",c)
# print(x.dtype)
```

```
plt.boxplot(c, vert=False)
plt.title("Boxplot of the sample after treating the outliers")
plt.xlabel("Sample")
```



REFERENCES & THANK YOU

- https://towardsdatascience.com/missing-value-handling-missing-data-types-a89c0d81a5bb
- https://towardsdatascience.com/7-ways-to-handle-missing-values-in-machine-learning-1a6326adf79e
- https://www.analyticsvidhya.com/blog/2021/05/detecting-and-treating-outliers-treating-the-odd-one-out/